A System for Answering Simple Questions in Multiple Languages

Anton Razzhigaev[1,2 *], **Mikhail Salnikov**[1 *], **Valentin Malykh**[3], **Pavel Braslavski**[4], **and Alexander Panchenko**[1,2]

[1] Skolkovo Institute of Science and Technology [2] Artificial Intelligence Research Institute
[3] ISP RAS Research Center for Trusted Artificial Intelligence [4] Ural Federal University

Abstract

Our research focuses on the most prevalent type of queries—*simple questions*—exemplified by questions like "What is the capital of France?". These questions reference an entity such as "France", which is directly connected (one hop) to the answer entity "Paris" in the underlying knowledge graph (KG). We propose a multilingual Knowledge Graph Question Answering (KGQA) technique that orders potential responses based on the distance between the question's text embeddings and the answer's graph embeddings. A system incorporating this novel method is also described in our work.

Through comprehensive experimentation using various English and multilingual datasets and two KGs — Freebase and Wikidata — we illustrate the comparative advantage of the proposed method across diverse KG embeddings and languages. This edge is apparent even against robust baseline systems, including seq2seq QA models, search-based solutions and intricate rule-based pipelines. Interestingly, our research underscores that even advanced AI systems like ChatGPT encounter difficulties when tasked with answering simple questions. This finding emphasizes the relevance and effectiveness of our approach, which consistently outperforms such systems. We are making the source code and trained models from our study publicly accessible to promote further advancements in multilingual KGQA.

1 Introduction

A knowledge graph (KG) is a collection of *subject–predicate–object* triples, for example ⟨Paris, capital_of, France⟩. Large KGs are valuable resources for many tasks, including question answering (QA) (Ji et al., 2022). Knowledge graph question answering (KGQA) is an active research area, as well as a popular application.

Even though all major web search engines implement KGQA capabilities – KG results can be easily recognized in their 'smart answers' – there are few operational KGQA research prototypes available online. A rare example is QAnswer (Diefenbach et al., 2020a), a rule-based KGQA system over Wikidata. There are also only a few free KGQA codebases available (Huang et al., 2019; Burtsev et al., 2018; Chen et al., 2021).

In this work, we focus on simple questions such as "What is the capital of France?". There exists an opinion that the task of answering such questions is nearly solved (Petrochuk and Zettlemoyer, 2018), but openly available systems are scarce and do not support multiple languages. Besides, their performance, as will be observed from our work, is still far from perfect even for models based on deep neural networks specifically pre-trained on QA data. In our work, our aim is to address these limitations of the prior art.

We developed a KGQA method M3M (multilingual triple match) based on text-to-graph embedding search. The key idea illustrated in Figure 1 is to combine a pre-trained multilingual language model for question representation and pre-trained graph embeddings that represent KG nodes and edges as dense vectors. In the training phase, we learn separate projections of the question text embeddings to the subject, predicate, and object of the KG triple corresponding to the question-answer pair. In the test phase, we first fetch a set of candidate KG triples based on the question's word n-grams and extract named entities to make the process more computationally efficient. Then, we rank candidate triples according to the sum of three cosine similarities – between the embeddings of the triple's components and respective projections of the question's embeddings. Finally, the object of the top-ranked triple is returned as an answer.

Our approach build upon Huang et al. (2019) expanding the method beyond a single KG and a single monolingual dataset. We experimented with the *de facto* standard English KGQA dataset

[*] The first two authors contributed equally.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 3: System Demonstrations, pages 524–537

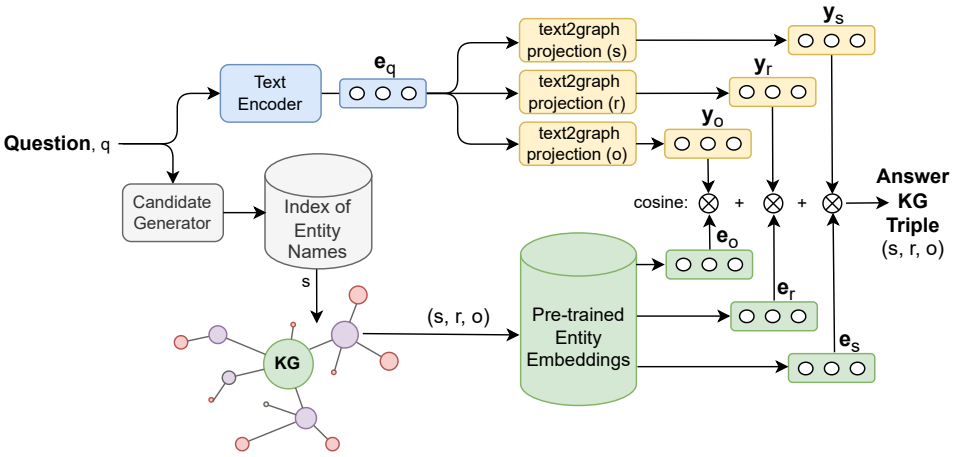

Figure 1: Workflow of M3M Knowledge Graph Question Answering system for simple questions.

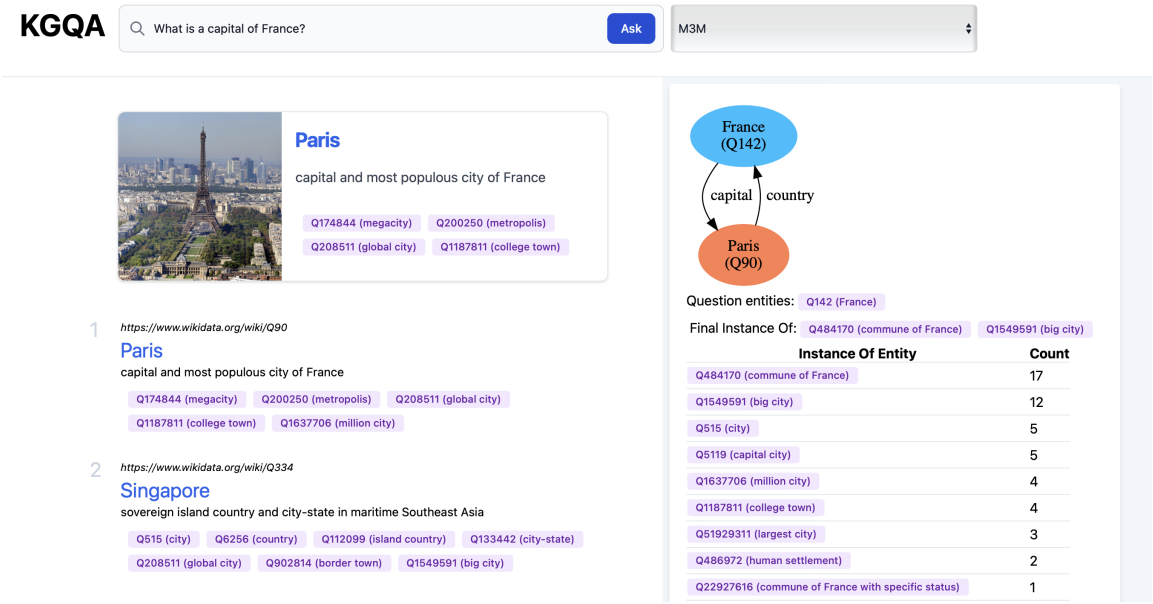

Figure 2: Graphical user interface of the KGQA system for answering one-hop questions.

SimpleQuestions, which is based on the now deprecated Freebase, to allow comparison with previous art. Moreover, we conducted experiments with several Wikidata-based datasets: SimpleQuestionsWd (a Wikidata mapping from the original benchmark), Russian/English RuBQ 2.0 dataset, as well as a recent Mintaka dataset covering nine languages. Our experiments demonstrate the applicability of the proposed method in different KGs and languages.

Our online demo (see Figure 2) implements two KGQA methods: (1) a T5 model fine-tuned on QA data and (2) our approach dubbed M3M based on embedding search. We believe that a combination of an online demo, publicly available code, as well as evaluation results on several datasets will contribute to future developments in the field of multilingual KGQA. To summarize, our contributions are the following:

- A novel multilingual approach to one-hop KGQA, which compares favorably to strong baselines, such as T5 QA system, and previous embedding-based methods on a battery of benchmarks.

- Open implementation of an online system for one-hop QA over Wikidata knowledge graph. We also release pre-trained models and provide an API making seamless integration into NLP applications possible.[1]

[1]Source code, link to the demo and video demonstration: https://github.com/s-nlp/m3m

2 Related Work

There are two main directions in KGQA research: semantic-parsing- and retrieval-based approaches. Semantic parsing seeks to map a natural language question to a meaning representation (e.g. SPARQL query language) that can be directly executed against a KG. Gu et al. (2022) provide a comprehensive survey of semantic parsing methods in KGQA. They also point out that KG schema richness (diversity of entity and relation types), irregularities and incompleteness of KG, as well as its large size make the application of semantic parsing methods to KGQA especially challenging.

The latter line of research follows information retrieval's notion of proximity between the query and sought entity. Early methods relied on the lexical match of the question and the KG elements (entity labels, descriptions, and relation names) (Balog, 2018). In their pioneering work Bordes et al. (2014) proposed to embed question words and KG components in a shared space, with question representations close to the answers' ones. This approach was further developed in a series of studies (Dai et al., 2016; Lukovnikov et al., 2017; Huang et al., 2019; Lukovnikov et al., 2019). Notably, more recent approaches leverage existing knowledge graph embeddings that have been used for a variety of KG tasks such as link prediction and entity classification (Wang et al., 2017) and pre-trained language models such as BERT applicable to a multitude of NLP tasks (Devlin et al., 2019). The embedding-based approach can be extended to complex KGQA (Saxena et al., 2020). A comprehensive overview of KGQA methods can be found in recent surveys (Chakraborty et al., 2021; Lan et al., 2021) and a monograph (Roy and Anand, 2021).

Large language models (LLMs) accumulate not only linguistic, but also factual knowledge, and thus can be considered as alternatives to KGs. Petroni et al. (2019) showed that BERT (a rather small model by today's standards) possessed some relational knowledge without fine-tuning. Roberts et al. (2020) experimented with the T5 model (Raffel et al., 2020) as an open-domain QA system. In particular, after fine-tuning on question-answer pairs, at inference time, the model was given a question and returned an answer without accessing any external knowledge source. The results were quite encouraging; the model with "salient span masking" (SSM) pre-training proposed by Guu et al. (2020) performed best. Mallen et al. (2022) found that LLMs memorise knowledge about common entities but fall short in the case of less common ones. They proposed to complement LLM-based QA with zero-shot and few-shot prompting with adaptive evidence retrieval. Tan et al. (2023) evaluate QA capabilities of FLAN-T5 (Chung et al., 2022) and several GPT versions (GPT3, GPT3.5, and chatGPT) in eight KGQA datasets. The results show that LLMs are yet quite far beyond SotA; chatGPT outperforms its contenders.

Multilingual KGQA has been relatively rarely addressed in the literature. Proposed solutions include question translation (Perevalov et al., 2022a), language-agnostic translation of syntactic parses into meaning representations (Reddy et al., 2017; Hakimov et al., 2017), and unsupervised induction of bilingual lexicons for word-level translation of training examples (Zhou et al., 2021). We hope that the advent of larger multilingual datasets with Wikidata annotations (Longpre et al., 2021; Sen et al., 2022) will stimulate interest in this timely topic.

3 M3M: Multilingual Triple Match QA

In this section, we describe the core of our contribution: a novel method for one-hop question answering over a knowledge graph. The proposed method dubbed M3M for Multilingual Triple Match relies on matching independently KGs subjects, objects, and predicates with the questions' representation. Overall, the approach can be considered as an improved version of the method by Huang et al. (2019) with which we perform extensive comparisons in the remainder of the paper among other baselines.

3.1 Question Answering Method

The architecture of the M3M model is illustrated in Figure 1. M3M uses multilingual BERT (Devlin et al., 2019) as question encoder. mBERT is a case-sensitive "base" 12-layer Transformer model that was trained with a masked language model objective on a collection of 104 Wikipedias. We encode the question q with 768-dimensional mBERT-based embeddings (M_{enc}) averaging the last hidden states of all tokens $e_q = M_{enc}(q)$. To project these representations to graph embeddings, we use three 2-layered perceptrons with ELU as an activation function. These three perceptrons are used to predict the embeddings of the object, relation, and

subject of the answer triple. We train four models simultaneously with AdamW optimiser with default parameters and $lr = 1e-4$ for 10 epochs. We use MSE loss between the predicted and ground truth graph embeddings.

At the test phase we first generate a set of candidate triples. We extract named entities from the question using an off-the-shelf NER tool, as well as all word uni-, bi-, and trigrams if there are no named entities found. We use SpaCy[2] as the base entity detection model and lemmatizer. After extracting all candidate entity mentions from the question, we search the collection of KG labels and retain all entities containing at least one of the extracted question parts as a substring in their labels. On average, the list contains ∼300 candidates.

Then, we map the question embedding e_q to the KG embedding space with three models (M_s, M_r, M_o), thus obtaining three projected embeddings of the question: e_s, e_r, e_o. We calculate the corresponding cosine similarities between the projected embeddings and KG embeddings of the candidate triples. Then, we sum these three cosine similarities, obtaining the answer score of each candidate triple. As graph embeddings are normalised and the model was trained with MSE loss, a sum of cosine similarities is an appropriate choice for a search metric. The QA system returns a ranked list of triples with the corresponding scores. Finally, the object of the highest-ranked triple is considered the answer. The overall schema of the proposed method for one-hop question answering is summarised in Algorithm 1.

3.2 System Implementation

The demo web application illustrated in Figure 2 is implemented using FastAPI framework.[3] It contains a Web application backend that can be used with any KGQA system over Wikidata. The application's frontend communicates with an asynchronous Web API (see Swagger documentation in Figure 7 in Appendix). The end user communicates with each of the provided pipelines by sending an HTTP POST request to **/pipelines/model** with a JSON object containing the question in the "text" field. Every pipeline responds to the end user with a JSON object containing an "answers" field with a sorted array of Wikidata identifiers. In addition to the "answers", the pipeline responses can include additional information, for example, /pipelines/m3m returns "scores" and "uncertainty". The entire KGQA system can be launched on any platform that supports Docker.[4] This demo can be easily integrated into other applications using the API endpoints provided.

We employ a conventional user interface for our Web demo – the KGQA system returns a ranked list of results along with links to Wikidata. In addition, the system response provides a visualization of the intermediate stages and sub-graphs corresponding to the questions, which helps to better understand the results. The core M3M method is based on a deeply reworked implementation of QA method firstly mentioned in (Chekalina et al., 2022) featuring various optimizations related to (1) effective candidate selection, and (2) retrieval of candidates and triples.

Algorithm 1 M3M method for one-hop KGQA

Input: q – question, $\mathcal{G}$ – KG, $\mathbf{E}$ – KG embeddings, M_{enc} – text encoder, M_s, M_r, M_o – projection modules, NER – subject candidates extractor
Output: $\langle o_a, r_a, s_a \rangle$ – triple with subject answer entity s_a

```
e_q = M_enc(q)
A=[]                                  ▷ Initialize answer-candidates
S=[]                                  ▷ Initialize scores
C=[]                                  ▷ Initialize entity-candidates
candidates = NER(q)
for entity in G do
    if entity.name in candidates then C.append(entity)
for entity in C do
    for relation in entity.relations do
        s = entity.id; r = relation.id; o = entity[r]
        A.append(⟨s, r, o⟩)
        e_s = E[s]; e_r = E[r]; e_o = E[o]
        y_s = M_s(e_q); y_r = M_r(e_q); y_o = M_o(e_q)
        score = cos(e_o, y_o) + cos(e_r, y_r) + cos(e_s, y_s)
        S.append(score)
ind = argmax(S)
⟨s_a, r_a, o_a⟩ = A[ind]
return ⟨s_a, r_a, o_a⟩
```

4 Experiments

In this section, we present the benchmarking of the presented system on several one-hop KGQA datasets for Freebase and Wikidata knowledge graphs. The method is compared to strong baselines. Both the datasets and the baselines are described below.

4.1 Data

Knowledge graphs. Freebase knowledge graph (Bollacker et al., 2008), launched in 2007, was a pioneering project that was developed to a large extent by community members. In 2010

[2]

[3] https://fastapi.tiangolo.com

[4] https://www.docker.com

Freebase was acquired by Google, and in 2015 it was discontinued as an independent project. The 'frozen' knowledge base dump was made publicly available and was also incorporated into Wikidata (Tanon et al., 2016). Nowadays Wikidata (Vrandecic and Krötzsch, 2014) is the largest and most up-to-date open KG. At the time of writing, Wikidata contains more than 100 million unique entities, 7,500 relation types, and 12.6 billion statements.[5] Wikidata items are provided with labels and descriptions in multiple languages. Many KGQA datasets have been using Freebase as an underlying knowledge graph. However, we can observe an increasing uptake of Wikidata as a basis for KGQA research. Wikidata underlies recent English (Cao et al., 2022; Dubey et al., 2019), as well as several multilingual datasets (Longpre et al., 2021; Rybin et al., 2021; Sen et al., 2022; Perevalov et al., 2022b). In our experiments, we use Freebase2M (FB2M) (Bordes et al., 2015) and Wikidata8M (Korablinov and Braslavski, 2020)[6] KG snapshots containing 2M and 8M entities, respectively.

KG embeddings. In our experiments, we use TransE KG embeddings that represent both entities and relations as vectors in the same space (Bordes et al., 2013). For KG triples $\langle s, p, o \rangle$ corresponding embeddings are expected to allow the following 'translation': $\mathbf{s}+\mathbf{p} \approx \mathbf{o}$. We conducted experiments on two TransE models: (1) TransE embeddings with dim=250 based on Freebase2M snapshot provided by Huang et al. (2019); and (2) TransE embeddings with dim=200 trained on the full Wikidata using Pytorch-BigGraph framework (Lerer et al., 2019).

QA datasets. In contrast to open-domain question answering and machine reading comprehension, there are rather few studies devoted to the multilingual KGQA (Perevalov et al., 2022a). This can partly be explained by the lack of data: until recently, KGQA datasets were almost entirely English, and rare exceptions such as QALD (Perevalov et al., 2022b) contained only hundreds of questions, which is rather scarce for training modern data-hungry models. Larger-scale datasets with KG annotations such as MKQA (Longpre et al., 2021) and Mintaka (Sen et al., 2022) have appeared only recently.

[5] https://www.wikidata.org/wiki/Property:P10209

[6] https://zenodo.org/record/3751761

To be able to compare our method with its predecessors, we use SimpleQuestions (Bordes et al., 2015). The dataset consists of 108,442 questions formulated by English speakers using Freebase triples as prompts. The question usually mentions the triple's subject and predicate, while the object is the expected answer. Note that by design the dataset contains many ambiguous questions having multiple answers in the KG, while only one answer is considered correct, see analysis in (Petrochuk and Zettlemoyer, 2018; Wu et al., 2020). Further, we make use of SimpleQuestionsWd, a mapping of SimpleQuestions to Wikidata (Diefenbach et al., 2017). The dataset contains 16,414/4,751 questions in train/test sets, respectively. We filtered the dataset and retained only questions, whose triples are present in Wikidata8M which resulted in 8,327 train and 2,438 test questions. RuBQ (Rybin et al., 2021) is a KGQA data set that contains 2,910 Russian questions of different types along with their English translations. Due to its limited size, the official RuBQ split has only development and test subsets. For our experiments, we kept only 1-hop questions that constitute the majority of the dataset. Mintaka (Sen et al., 2022) is a recently published dataset that contains 20k questions of different types translated into eight languages. For our experiments, we took only *generic* questions, whose entities are one hop away from the answers' entities in Wikidata, which resulted in 1,236/181/340 train/dev/test questions in each language, respectively.[7] Table 1 summarizes the data we used for training and evaluation.

Dataset	KG	Lang.	Train	Test
SimpleQuestions (SQ)	FB	en	75,910	21,687
SimpleQuestionsWd	WD	en	8,327	2,438
RuBQ	WD	ru, en	300	1,186
Mintaka-Simple	WD	mult	1,236	340

Table 1: Datasets in our experiments (FB – Freebase, WD – Wikidata).

4.2 Baselines

We compare our system with three groups of methodologically diverse state-of-the-art QA systems: based on rule-based approach, relying on

[7] These questions are not necessarily *simple* in terms of SimpleQuestions dataset. E.g. both entities (*Heath Ledger, The Dark Knight*) in the question *Who did Heath Ledger play as in the movie The Dark Knight?* are one hop away from the answer entity *Joker*; both triples are essential to provide a correct answer.

embedding search similar to our approach, and generative neural models, e.g. sequence-to-sequence.

QAnswer is a rule-based multilingual QA system proposed by Diefenbach et al. (2020b). It returns a ranked list of Wikidata identifiers as answers and a corresponding SPARQL query. We use QAnswer API in our experiments.[8]

KEQA: Knowledge Embedding based Question Answering. There are no published results on SimpleQuestions aligned with Wikidata KG which is why we adopt the official implementation of KEQA (Huang et al., 2019) – an open-source embedding-based KGQA solution to SimpleQuestionsWd benchmark. It was initially trained and evaluated on Freebase embeddings. To the best of our knowledge, there are no open-sourced KGQA models with better performance than KEQA. We use this model as the main baseline on SimpleQuestionsWd and original SimpleQuestions benchmarks. To make a comparison with KEQA on the SimpleQuestionsWd test set more fair, we retrain it on PTBG-Wikidata embeddings. We use the official implementation with provided hyperparameters and an internal validation mechanism.[9] As SimpleQuestionsWd is a subset of the original SimpleQuestions, we evaluate Freebase-pretrained KEQA on this dataset as well (taking into account the corresponding entity mapping).

T5-based QA system. Question answering can also be addressed as a seq2seq task. To provide a comparison with this type of approaches, we conducted experiments with T5, an encoder-decoder transformer-based model pre-trained on a multi-task mixture of unsupervised and supervised tasks (Raffel et al., 2020). T5 works well on a variety of tasks out-of-the-box by prepending a prefix to the input corresponding to each of the tasks. To answer English questions, we used T5 model fine-tuned on a large *NaturalQuestions* dataset (Roberts et al., 2020). For other languages, we fine-tuned mT5-xl model (Xue et al., 2021) on Mintaka Simple.

In addition, we carried out experiments employing the Flan-T5-xl (Chung et al., 2022) model, a recent development trained on a diverse mixture of tasks. We evaluated this model in two distinct setups using the Mintaka dataset: firstly in a zero-shot setting, utilizing the prompt "Question: question Answer:", and secondly, by separately fine-tuning the model on the training data for each individual language.

GPT-3 has gained recognition for its impressive performance in both few-shot and zero-shot contexts (Brown et al., 2020), excelling in a vast array of benchmarks. A recent study (Chung et al., 2022) evaluate different GPT versions on complex KG questions. However, the experiments don't include datasets in our study. To address this oversight and offer a comparative baseline for our system, we subjected the GPT-3 model (davinci-003) to the SimpleQuestionsWd and RuBQ 2.0 benchmarks. Detailed information on the generation parameters and prompts can be found in the Appendix.

ChatGPT stands as one of the leading systems in the field of Natural Language Processing (NLP), demonstrating capabilities for intricate reasoning and extensive factual knowledge (OpenAI, 2023). We evaluated this system (GPT-3.5-turbo-0301) using the RuBQ 2.0 and SimpleQuestionsWd benchmarks. Specifics about the prompts and generation parameters are available in the Appendix.

4.3 Experimental Setup

To compare our algorithm with baselines, we use the Accuracy@1 metric i.e. correctness of the first retrieved result. The answer of a QA system to an answerable question is considered correct if its object matches the answer in terms of Wikidata id or just by a label string.

It is essential to acknowledge that sequence-to-sequence (seq2seq) models yield a string instead of a knowledge graph ID, which may pose a challenge during evaluation. To mitigate this, we apply specific transformations to the responses produced by seq2seq systems. These include converting the text to lowercase and eliminating any leading and trailing spaces. This transformation process is also applied to label-aliases representing the actual answers present in the RuBQ and Mintaka datasets.

Regarding the SimpleQuestionsWd dataset, we procure aliases for the correct answers via the Wikidata API.[10] We then determine the accuracy of the seq2seq model's prediction by checking for an exact match between the predicted string and one of the aliases.

[8] https://qanswer-frontend.univ-st-etienne.fr

[9] https://github.com/xhuang31/KEQA_WSDM19

[10] https://pypi.org/project/Wikidata

Model	SQ	SQ-WD	RuBQ-ru	RuBQ-en
QAnswer (Diefenbach et al., 2020a)	–	33.31	30.80	32.30
T5-11b-ssm-nq, fine-tuned (Roberts et al., 2020)	–	20.40	–	42.75
ChatGPT – GPT-3.5-turbo	–	17.75	26.99	30.12
GPT-3 – davinci-003	–	28.51	18.10	34.20
KEQA (Huang et al., 2019) – TransE FB2M	75.40	40.48	–	–
KEQA (Huang et al., 2019) – TransE PTBG	–	48.89	–	33.80
M3M (Ours) – TransE FB2M	**76.90** ± 0.30	–	–	–
M3M (Ours) – TransE PTBG	–	**53.50** ± 0.30	**48.40** ± 0.30	**49.50** ± 0.30

Table 2: Comparison of M3M system with KGQA baselines in terms of Accuracy@1 for monolingual one-hop QA datasets. The best scores are highlighted. M3M scores are averages over models trained with five random seeds.

Model	en	es	de	ar	fr	pt	it	hi	avg
mT5-xl, fine-tuned (Xue et al., 2021)	20.8	19.5	19.3	12.6	19.7	18.3	20.9	9.7	17.6
FlanT5-xl, fine-tuned (Chung et al., 2022)	**35.3**	22.0	23.3	0.2	**25.0**	24.0	**25.5**	0	19.41
FlanT5-xl, zero-shot (Chung et al., 2022)	14.7	6.5	7.6	0	0.7	0.9	0.9	0	3.19
M3M (Ours) – TransE PTBG	26.0	**26.1**	**25.0**	**24.1**	**25.0**	**24.7**	25.3	**24.1**	**25.0**

Table 3: Results on Mintaka-Simple dataset (one-hop questions) for models trained simultaneously on all languages.

It is noteworthy to mention that in the Mintaka-Simple test set, about a half of the answers don't have labels in Hindi.

4.4 Results

Table 2 contains the results of our M3M model and several baselines on two versions of the SimpleQuestions dataset and two versions of the RuBQ dataset. Specifically, for the RuBQ dataset, we detail the outcomes derived from testing both Russian and English language queries.

Interestingly, ChatGPT, despite being recognized as a more sophisticated system, exhibits a weaker performance on factoid questions compared to GPT-3. Upon conducting a concise manual error analysis, we observed that ChatGPT frequently dismissed queries with responses such as "Answer is unknown", or sought supplemental information. We suggest that this behavior may be a consequence of the system's alignment with human feedback, implemented to limit the model's tendency for generating ungrounded or 'hallucinated' responses. However, it is plausible that a more refined prompt design could address this issue and enhance the system's performance on such questions. Nonetheless, this exploration extends beyond the scope of our current research and is suggested as an avenue for further investigation.

Table 3 features the results obtained on the Mintaka-Simple dataset, providing an opportunity to evaluate the mT5, Flan-T5 and M3M models. This table highlights the multilingual capabilities exhibited by both the generative and the KG-retrieval approaches. An analysis of these results reveals that our model's performance is markedly stable across languages, indicating a lesser dependence on the language relative to the seq2seq approach. Our model manifests exceptional performance on one-hop simple questions and achieves a new state-of-the-art on the RuBQ 2.0 (Russian) benchmark as well as on the English SimpleQuestionsWd dataset. These findings illustrate the superiority of KG-based models, outperforming both GPT-3 and ChatGPT by a considerable margin.

5 Conclusion

In this study, we introduced M3M, a multilingual model, along with an open implementation, devised for one-hop knowledge base question answering. Our approach leverages the use of a multilingual text encoder and pre-trained KG embeddings, which are aligned using a triple projection method of a question to subject/relation/object of KG triple to facilitate efficient answer search in the embedding space.

For simple questions, our system not only outperforms previous strong alternatives, including rule-based approaches, embeddings-based similarity search, and pre-trained sequence-to-sequence neural models, but also excels when compared to advanced AI models like ChatGPT. These comparative results were drawn from a comprehensive battery of one-hop QA datasets, including both monolingual and multilingual data.

6 Limitations

While a large fraction of users' information needs may be fulfilled by answering simple questions, the main limitation of the proposed system is that in the current implementation, it can be applied only to one-hop KG questions. As it may be not obvious to figure out beforehand which question is one-hop and which is multi-hop in a KG special classifiers or uncertainty estimation techniques should be ideally combined with the proposed system to not let the system answer questions it is not designed to answer in the first place. At the same time, our preliminary experiments with training classifiers of question type based on Mintaka data show promising results, suggesting that such classifier effectively could be created and used in real deployments in conjunction with the proposed system.

In terms of computational efficiency, communication with a knowledge graph can be a bottleneck if based on a public SPARQL endpoint with query limits but could be substantially sped up using an in-house SPARQL engine or using indexing of triples with appropriate data structures. However, in the latter case, a mechanism for updating such structures is requires to keep system answers up to date.

7 Ethical Statement

QA systems built on top of large pretrained neural models, such as those described in this paper, may transitively reflect biases available in the training data potentially generating stereotyped answers to questions. It is therefore recommended in production (as compared to research settings) to use a special version of debiased pre-trained neural models and/or deploy a special layer of debiasing systems around the proposed methodology.

Acknowledgements

Pavel Braslavski's work was supported in part by the Ministry of Science and Higher Education of the Russian Federation (project 075-02-2023-935). The work of Valentin Malykh was supported by a grant for research centers in the field of artificial intelligence, provided by the Analytical Center for the Government of the Russian Federation in accordance with the subsidy agreement (agreement identifier 000000D730321P5Q0002) and the agreement with the Ivannikov Institute for System Programming of the Russian Academy of Sciences dated November 2, 2021 No. 70-2021-00142.

References

Krisztian Balog. 2018. *Entity-Oriented Search*, volume 39 of *The Information Retrieval Series*. Springer.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, page 1247–1250, New York, NY, USA. Association for Computing Machinery.

Antoine Bordes, Sumit Chopra, and Jason Weston. 2014. Question answering with subgraph embeddings. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 615–620. ACL.

Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. 2015. Large-scale simple question answering with memory networks. *CoRR*, abs/1506.02075.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 2787–2795.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Mikhail S. Burtsev, Alexander V. Seliverstov, Rafael Airapetyan, Mikhail Y. Arkhipov, Dilyara Baymurzina, Nickolay Bushkov, Olga Gureenkova, Taras Khakhulin, Yuri Kuratov, Denis Kuznetsov, Alexey Litinsky, Varvara Logacheva, Alexey Lymar, Valentin Malykh, Maxim Petrov, Vadim Polulyakh, Leonid Pugachev, Alexey Sorokin, Maria Vikhreva, and Marat Zaynutdinov. 2018. Deeppavlov: Open-source library for dialogue systems. In *Proceedings of ACL 2018, Melbourne, Australia, July 15-20, 2018, System Demonstrations*, pages 122–127. Association for Computational Linguistics.

Shulin Cao, Jiaxin Shi, Liangming Pan, Lunyiu Nie, Yutong Xiang, Lei Hou, Juanzi Li, Bin He, and Hanwang Zhang. 2022. KQA pro: A dataset with explicit compositional programs for complex question answering over knowledge base. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6101–6119, Dublin, Ireland. Association for Computational Linguistics.

Nilesh Chakraborty, Denis Lukovnikov, Gaurav Maheshwari, Priyansh Trivedi, Jens Lehmann, and Asja Fischer. 2021. Introduction to neural network-based question answering over knowledge graphs. *WIREs Data Mining Knowl. Discov.*, 11(3).

Viktoriia Chekalina, Anton Razzhigaev, Albert Sayapin, Evgeny Frolov, and Alexander Panchenko. 2022. MEKER: Memory efficient knowledge embedding representation for link prediction and question answering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 355–365, Dublin, Ireland. Association for Computational Linguistics.

Shuang Chen, Qian Liu, Zhiwei Yu, Chin-Yew Lin, Jian-Guang Lou, and Feng Jiang. 2021. ReTraCk: A flexible and efficient framework for knowledge base question answering. In *Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL 2021 - System Demonstrations, Online, August 1-6, 2021*, pages 325–336. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416.

Zihang Dai, Lei Li, and Wei Xu. 2016. CFO: Conditional focused neural question answering with large-scale knowledge bases. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 800–810, Berlin, Germany. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Dennis Diefenbach, Andreas Both, Kamal Singh, and Pierre Maret. 2020a. Towards a question answering system over the semantic web. *Semantic Web*, 11(3):421–439.

Dennis Diefenbach, José M. Giménez-García, Andreas Both, Kamal Singh, and Pierre Maret. 2020b. Qanswer KG: designing a portable question answering system over RDF data. In *The Semantic Web - 17th International Conference, ESWC 2020, Heraklion, Crete, Greece, May 31-June 4, 2020, Proceedings*, volume 12123 of *Lecture Notes in Computer Science*, pages 429–445. Springer.

Dennis Diefenbach, Thomas Pellissier Tanon, Kamal Deep Singh, and Pierre Maret. 2017. Question answering benchmarks for wikidata. In *Proceedings of the ISWC 2017 Posters & Demonstrations and Industry Tracks co-located with 16th International Semantic Web Conference (ISWC 2017), Vienna, Austria, October 23rd - to - 25th, 2017*, volume 1963 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Mohnish Dubey, Debayan Banerjee, Abdelrahman Abdelkawi, and Jens Lehmann. 2019. Lc-quad 2.0: A large dataset for complex question answering over wikidata and dbpedia. In *The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part II*, volume 11779 of *Lecture Notes in Computer Science*, pages 69–78. Springer.

Yu Gu, Vardaan Pahuja, Gong Cheng, and Yu Su. 2022. Knowledge base question answering: A semantic parsing perspective. *CoRR*, abs/2209.04994.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR.

Sherzod Hakimov, Soufian Jebbara, and Philipp Cimiano. 2017. AMUSE: multilingual semantic parsing for question answering over linked data. In *The Semantic Web - ISWC 2017 - 16th International Semantic Web Conference, Vienna, Austria, October 21-25, 2017, Proceedings, Part I*, volume 10587 of *Lecture Notes in Computer Science*, pages 329–346. Springer.

Xiao Huang, Jingyuan Zhang, Dingcheng Li, and Ping Li. 2019. Knowledge graph embedding based question answering. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, WSDM 2019, Melbourne, VIC, Australia, February 11-15, 2019*, pages 105–113. ACM.

Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip S. Yu. 2022. A survey on knowl-

edge graphs: Representation, acquisition, and applications. *IEEE Trans. Neural Networks Learn. Syst.*, 33(2):494–514.

Vladislav Korablinov and Pavel Braslavski. 2020. RuBQ: A Russian dataset for question answering over Wikidata. In *The Semantic Web - ISWC 2020 - 19th International Semantic Web Conference, Athens, Greece, November 2-6, 2020, Proceedings, Part II*, volume 12507 of *Lecture Notes in Computer Science*, pages 97–110. Springer.

Yunshi Lan, Gaole He, Jinhao Jiang, Jing Jiang, Wayne Xin Zhao, and Ji-Rong Wen. 2021. A survey on complex knowledge base question answering: Methods, challenges and solutions. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pages 4483–4491. ijcai.org.

Adam Lerer, Ledell Wu, Jiajun Shen, Timothée Lacroix, Luca Wehrstedt, Abhijit Bose, and Alex Peysakhovich. 2019. Pytorch-BigGraph: A large scale graph embedding system. In *Proceedings of Machine Learning and Systems 2019, MLSys 2019, Stanford, CA, USA, March 31 - April 2, 2019*. mlsys.org.

Shayne Longpre, Yi Lu, and Joachim Daiber. 2021. MKQA: A linguistically diverse benchmark for multilingual open domain question answering. *Trans. Assoc. Comput. Linguistics*, 9:1389–1406.

Denis Lukovnikov, Asja Fischer, and Jens Lehmann. 2019. Pretrained transformers for simple question answering over knowledge graphs. In *The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part I*, pages 470–486.

Denis Lukovnikov, Asja Fischer, Jens Lehmann, and Sören Auer. 2017. Neural network-based question answering over knowledge graphs on word and character level. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, pages 1211–1220. ACM.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi, and Daniel Khashabi. 2022. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. *CoRR*, abs/2212.10511.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Aleksandr Perevalov, Andreas Both, Dennis Diefenbach, and Axel-Cyrille Ngonga Ngomo. 2022a. Can machine translation be a reasonable alternative for multilingual question answering systems over knowledge graphs? In *WWW '22: The ACM Web Conference 2022, Virtual Event, Lyon, France, April 25 - 29, 2022*, pages 977–986. ACM.

Aleksandr Perevalov, Dennis Diefenbach, Ricardo Usbeck, and Andreas Both. 2022b. Qald-9-plus: A multilingual dataset for question answering over dbpedia and wikidata translated by native speakers. In *16th IEEE International Conference on Semantic Computing, ICSC 2022, Laguna Hills, CA, USA, January 26-28, 2022*, pages 229–234. IEEE.

Michael Petrochuk and Luke Zettlemoyer. 2018. SimpleQuestions nearly solved: A new upperbound and baseline approach. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 554–558, Brussels, Belgium. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Siva Reddy, Oscar Täckström, Slav Petrov, Mark Steedman, and Mirella Lapata. 2017. Universal semantic parsing. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 89–101, Copenhagen, Denmark. Association for Computational Linguistics.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Rishiraj Saha Roy and Avishek Anand. 2021. *Question Answering for the Curated Web: Tasks and Methods in QA over Knowledge Bases and Text Collections*. Synthesis Lectures on Information Concepts, Retrieval, and Services. Morgan & Claypool Publishers.

Ivan Rybin, Vladislav Korablinov, Pavel Efimov, and Pavel Braslavski. 2021. RuBQ 2.0: An innovated russian question answering dataset. In *The Semantic Web - 18th International Conference, ESWC 2021*, pages 532–547.

Apoorv Saxena, Aditay Tripathi, and Partha Talukdar. 2020. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4498–4507, Online. Association for Computational Linguistics.

Priyanka Sen, Alham Fikri Aji, and Amir Saffari. 2022. Mintaka: A complex, natural, and multilingual dataset for end-to-end question answering. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1604–1619, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Yiming Tan, Dehai Min, Yu Li, Wenbo Li, Nan Hu, Yongrui Chen, and Guilin Qi. 2023. Evaluation of chatgpt as a question answering system for answering complex questions. *CoRR*, abs/2303.07992.

Thomas Pellissier Tanon, Denny Vrandecic, Sebastian Schaffert, Thomas Steiner, and Lydia Pintscher. 2016. From freebase to wikidata: The great migration. In *Proceedings of the 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, April 11 - 15, 2016*, pages 1419–1428. ACM.

Denny Vrandecic and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.

Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE Trans. Knowl. Data Eng.*, 29(12):2724–2743.

Zhiyong Wu, Ben Kao, Tien-Hsuan Wu, Pengcheng Yin, and Qun Liu. 2020. PERQ: predicting, explaining, and rectifying failed questions in KB-QA systems. In *WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020*, pages 663–671. ACM.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 483–498. Association for Computational Linguistics.

Yucheng Zhou, Xiubo Geng, Tao Shen, Wenqiang Zhang, and Daxin Jiang. 2021. Improving zero-shot cross-lingual transfer for multilingual question answering over knowledge graph. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5822–5834, Online. Association for Computational Linguistics.

A Parameter Settings and Prompts for OpenAI Models

The parameters for GPT-3 (davinci-003) and ChatGPT (GPT-3.5-turbo-0301) were configured to a temperature setting of 0.1, while the top_p for GPT-3 was set to 0.85. The prompts for both models, which are illustrated in Figure 3, were used respectively: for GPT-3 directly, and for ChatGPT, conveyed via the "user" field in the chat API interface.

```
Answer the following question with shortest string possible.
Follow the exactly same format (don't forget about punctuation):

Question: Who is the president of USA?
Answer: Joe Biden

Question: How many musicians are there in the Quartet?
Answer: 4

Question: {question}
Answer:
```

```
Ответь максимально коротко и ёмко на вопрос.
Ответ приводи в именительном падеже, при этом не забывай о пунктуации.
Если ответов на вопрос несколько, то приведи наиболее известный.
Вот несоклько примеров для понимания формата:

Question: Кто является президентом США?
Answer: Джо Байден

Question: Сколько музыкантов в квартете?
Answer: 4

Question: {question}
Answer:
```

Figure 3: Prompts employed for ChatGPT and GPT-3 in the English and Russian language benchmarks, respectively.

B Additional Illustrations of the Knowledge Graph Question Answering System

This section contains three illustrations of the graphical user interface with three additional questions as well as a demonstration of the API for integration into external applications.

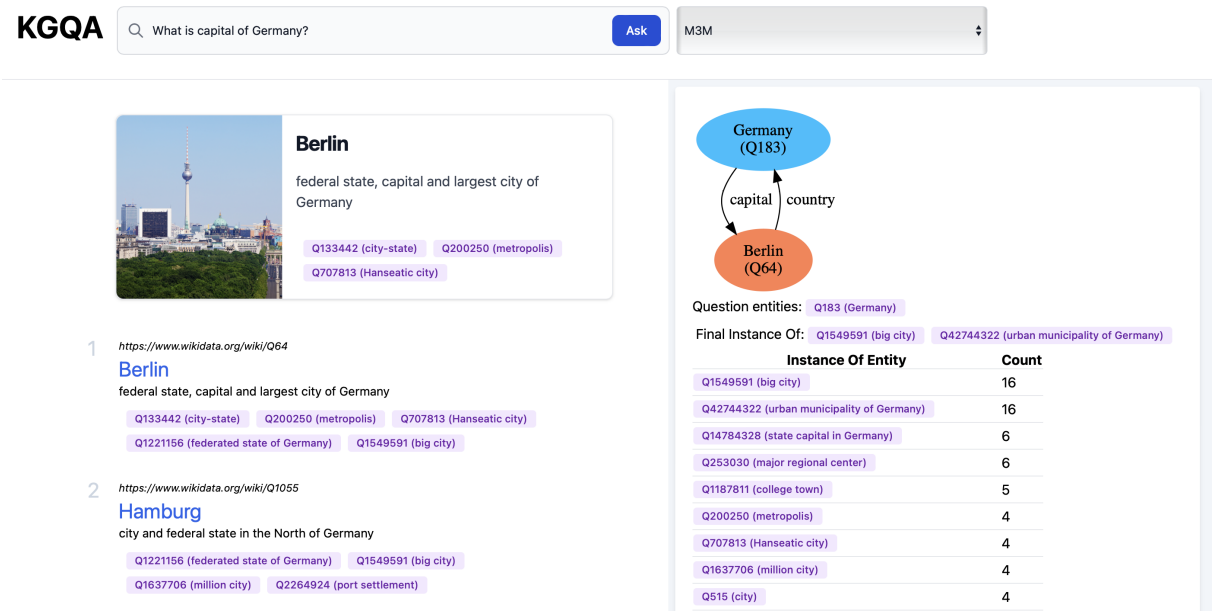

Figure 4: Graphical user interface of KGQA system for the one-hop question "What is capital of Germany?".

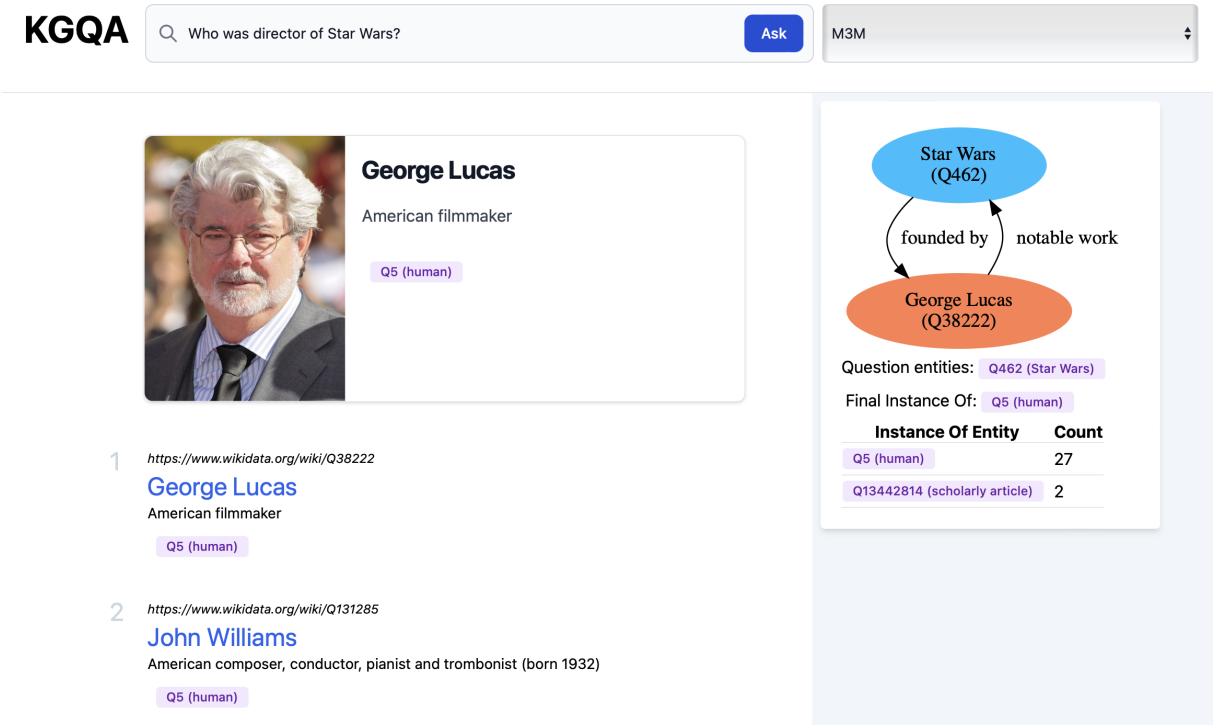

Figure 5: Graphical user interface of KGQA system for the one-hop question "Who was director of Star Wars?".

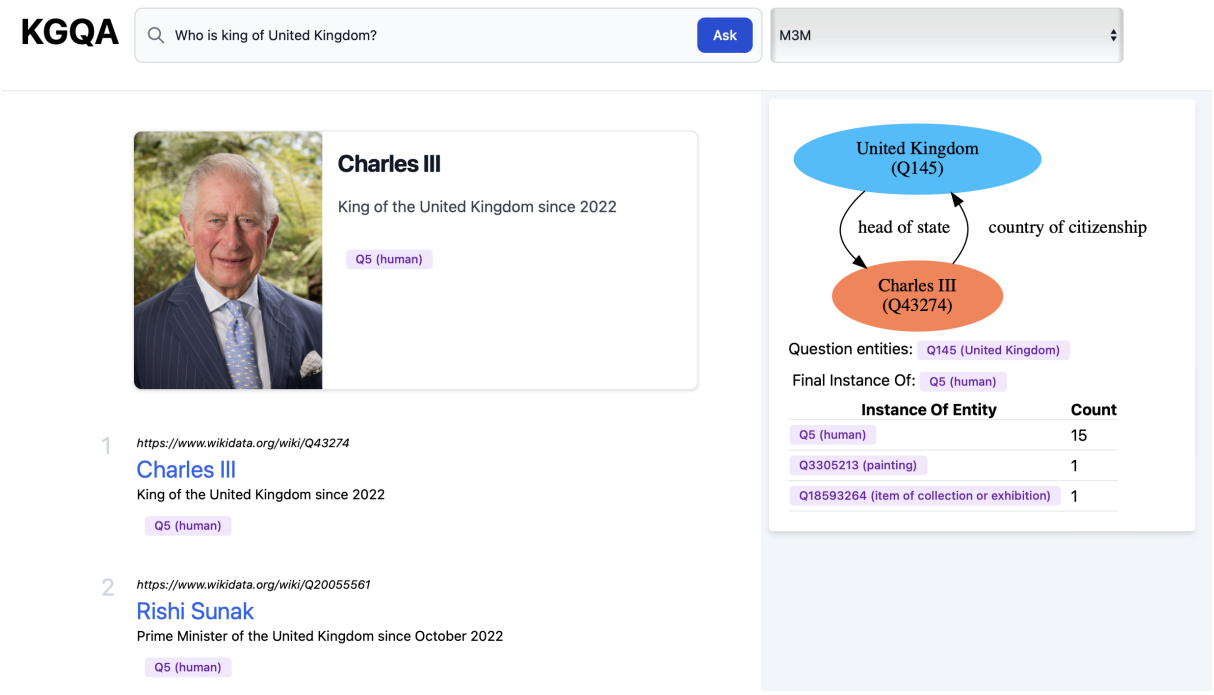

Figure 6: Graphical user interface of KGQA system for the one-hop question "Who is the king of United Kingdom?".

Figure 7: Swagger API for the developed system allowing for integration of the KGQA functionality into applications (e.g. “seq2seq” or “m3m” endpoints) as well as subcomponents, such as NER for questions or type selection.

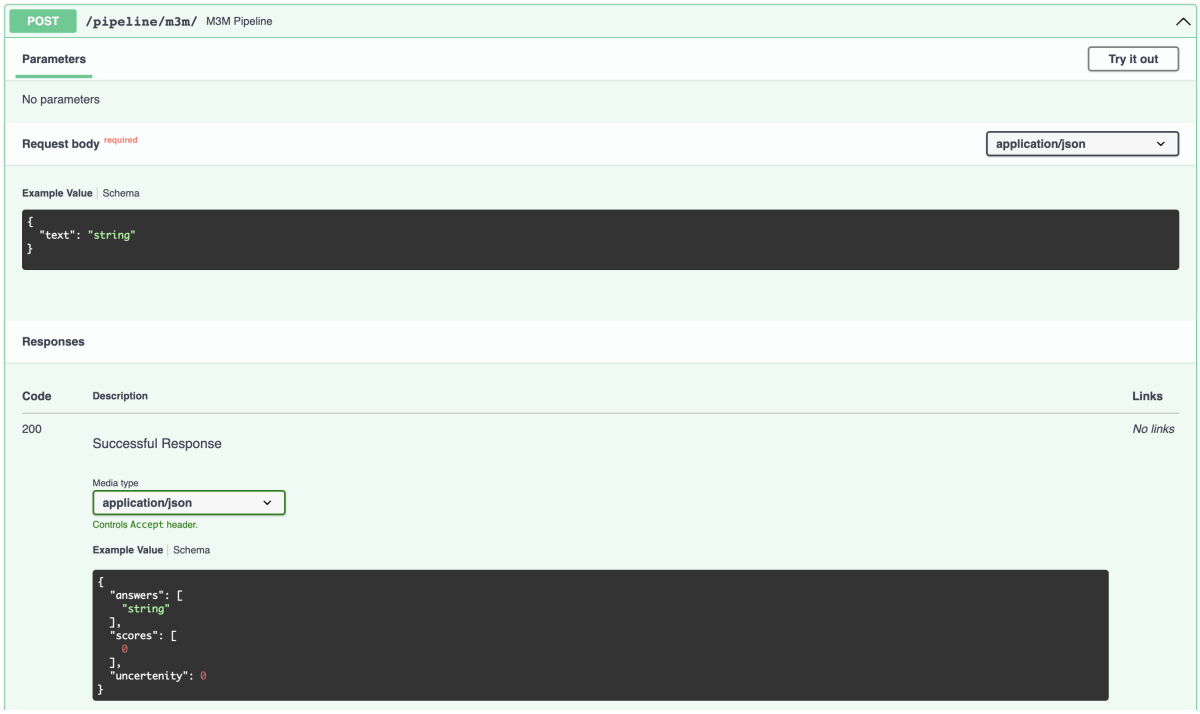

Figure 8: Swagger API of an individual endpoint: parameters for KGQA method are documented and can be called using a RESTful endpoint.